Direct and Indirect Approaches to Blockmodeling of Valued Networks in Terms of Regular Equivalence

Aleš Žiberna (University of Ljubljana)

Abstract

The aim of this paper is to compare methods for detecting and/or measuring regular equivalence in valued networks.

The methods compared can be divided into direct and indirect methods. The indirect methods considered here are all variants of the REGE algorithm (White, 2005). All direct methods have roots in Batagelj, Doreian, and Ferligoj (1992); however they have been adapted for valued networks (Žiberna, 2006b).

All approaches are compared on an empirical valued social network. Variants of the REGE algorithm are also compared on a few artificial valued networks in order to clarify their functionality. This is not necessary for direct approaches, which have clearly defined ideal blocks.

In discussion, advantages and disadvantages of both indirect and direct approaches are discussed.

Keywords: social networks, blockmodeling, generalized blockmodeling, REGE, regular equivalence, valued networks

Introduction

Regular equivalence has been studied extensively in the past. However, surprisingly little attention has been given to regular equivalence in (interval[1]) valued networks, especially in terms of its theoretical properties and definitions. This is even more surprising, since an algorithm for detecting regular equivalence (REGE) had also been developed for valued

[1] Values are assumed to be measured on at least an interval scale.

networks from the start (White, 1985a)[2]. This algorithm was also applied to valued networks (Smith and White, 1992).

The use of REGE for blockmodeling constitutes an indirect approach to blockmodeling. First REGE is used to compute a (dis)similarity matrix, which is then used as input to an appropriate clustering algorithm. White (2005) presents two versions of REGE, which date back to 1985.

However, these have never been properly compared neither among themselves nor to the direct approach for measuring regular equivalence introduced by Batagelj, Doreian, and Ferligoj (1992). Although this approach was originally developed for binary networks, it was extended to valued networks by Batagelj and Ferligoj (2000) and Žiberna (2006b). The aim of this paper is to compare all these approaches. In addition, four additional versions of REGE are used in the comparison. One of these is an older version of REGGE (itself a version of REGE[3]), while the other three were developed by modifying the three versions mentioned above. The R package `blockmodeling` (Žiberna, 2006a) was used to perform all the analysis presented in this paper.

The notation used in the remainder of the paper is presented in Section 2. The definitions of regular equivalence for binary and valued networks are given and discussed in Section 3. An alternative equivalence for valued networks, an *f*-regular equivalence, is also defined and discussed. Generalized blockmodeling, including binary, valued, homogeneity and implicit blockmodeling, is reviewed in Section 4. REGE (several variants) is described in Section 5. In Section 6, some properties of different versions of REGE are clarified through a set of

[2] The algorithm was probably modified after the year 1985; however this is the year written in the algorithm.

[3] In this text, the term REGE is used to for all versions of the REGE algorithm, while REGGE is used for the "classical" similarity version of REGE (White, 1985a). REGGE was in other works also often denoted as REGE.

artificial networks. All the approaches presented are applied to an empirical example in Section 7. The discussion follows in Section 8.

Notation

The notation used is:

- The network N = (U, R), where U is a set of all units $U = (u_1, u_2, \ldots, u_n)$ and R is the relation among these units $R \subseteq U \times U$. (The network can also have multiple relations N = ($U, R_1, R_2, \ldots, R_m$), where m is the number of relations.)
- In generalized blockmodeling, a relation R is usually represented by a valued matrix R with elements $[r_{ij}]$, where r_{ij} indicates the value (or weight) of the arc from unit i to unit j; $r: R \to \mathbb{R}$, $r_{ij} = \begin{cases} r(i,j), (i,j) \in R \\ 0, \text{otherwise} \end{cases}$
- C_i is a cluster of units.
- $\mathbf{C} = \{C_1, C_2, \ldots, C_n\}$ is a partition of the set U; $\bigcup_{i=1}^{n} C_i = U$; $C_i \cap C_i = 0, i \neq j$.
- Φ is a set of all feasible partitions.
- A partition $\mathbf{C}$ also partitions the relation R into blocks; $R(C_i, C_j) = R \cap C_i \times C_j$. Each such block consists of units belonging to clusters C_i and $C_{j,,}$ and all arcs leading from cluster C_i to cluster C_j. If $i = j$, a block $R(C_i, C_i)$ is called a diagonal block.
- Let $T(C_i, C_j)$ denote a set of all ideal blocks, corresponding to block $R(C_i, C_j)$. For definition of ideal blocks, see Žiberna(2006b).
- f is a function that assigns a real value to a valued vector of length n; f: $\mathbb{R}^n \to \mathbb{R}$. For example, this function can be *mean, maximum, sum,* etc.

Regular equivalence

Binary networks

One common type of equivalence is regular equivalence. "Regularly equivalent points (units) are connected in the same way to matching equivalents" (White and Reitz, 1983: 200). A more formal definition[4] is as follows (White and Reitz, 1983: 200):

If $N = (U, R)$ and $\equiv$ is an equivalence relation on U, then $\equiv$ is a regular equivalence if and only if for all $a, b, c \in U$, $a \equiv b$ implies:

(i) aRc implies there exists $d \in U$ such that bRd and $d \equiv c$;

(ii) cRa implies there exists $d \in U$ such that dRb and $d \equiv c$.

Batagelj, Doreian, and Ferligoj (1992: 125) proposed that if $C = \{C_i\}$ is a partition corresponding to a regular equivalence, then either $R(C_u , C_v)$ is null (empty) or it has a property that there exists at least one 1 (tie) in each of its rows and in each of its columns.

However, the answer to the question of whether this fully complies with the above definition by White and Reitz (1983: 200) depends on the interpretation of this definition. Obviously, Batagelj, Doreian, and Ferligoj (1992) and Borgatti and Everett (1992) understood it in a way that allows d in (i) and (ii) to be different units, while examining the REGGE and REGDI algorithms by White (1985a, b) shows that White intended that d represent the same unit in both (i) and (ii). The interpretation of Batagelj, Doreian, and Ferligoj (1992) and Borgatti and Everett (1992) is better suited to a generalized blockmodeling approach, because in this way

[4] In the definition, N as a network is used instead of G as a graph in the original definition in an attempt to establish common notation for this paper. U as a unit set is used instead of P as a point for the vertex set for the same reasons.

each block can be evaluated independently. Therefore, only this one is used for all direct methods considered in this text, since they are all based on generalized blockmodeling.

Regular equivalence evolved from a notion of *structural relatedness* introduced by Salier (1978: 78) based on Boyd's suggestions for a definition of structural equivalence. This relation is not equivalence, and it is not necessarily transitive and symmetric, as Salier noted. Salier's definition of structural relatedness is as follows (expressed in the notation used in this paper):

If $N = (U, R)$, then SR is a *structural relatedness* relation on U if and only if, for all $a, b, c \in U$, $aRSb$ implies that whenever there exists such c that cRa, there exists such $d \in U$ that dRb and $cSRd$.

As we can see, this is very similar to the definition of regular equivalence without list item (i). In addition, structural relatedness is defined as a relation, not as an equivalence relation. This means that it is not necessarily symmetric and transitive. Salier also states that this definition can be naturally generalized also to include outgoing ties (and multiple relations). In a case where both in- and outgoing ties are used, a symmetric[5] version of his definition (based on his idea for computing similarities based on in- and outgoing ties) would correspond to an understanding of the definition of regular equivalence as understood by Batagelj, Doreian, and Ferligoj (1992) and Borgatti and Everett (1992).

Valued networks

Regular equivalence for valued networks was first implemented in the REGGE algorithm (White, 2005; 1985a), which also implied a definition of regular equivalence for valued networks. Several versions of REGE exist, which are discussed in more detail in the section

[5] He did not actually suggest a symmetric version.

on REGE. Discussion is limited to versions of REGE designed for valued networks measured on at least an interval scale. All versions have a basic principle in common, which can be seen from the FORTAN program written by White (1985a) to implement the algorithm for measuring similarities of units in terms of regular equivalence introduced by White and Reitz (1983).

REGE evaluates the (degree of) equivalence of units *a* and *b* by trying to match every link[6] (taking values into account) of unit *a* by the most similar link of unit *b* to an equivalent / the most equivalent unit and vice versa (every link of unit *b* an equivalent / the most equivalent link of unit *a*).

If the best match (link) does not have equal (or equal or grater, depending on the version of REGE) values of ties, or if the other points (ends) of ties (from and to *a*; from and to *b*) are not completely equivalent, a (dis)similarity depends on the difference of the values (of ties) and on the (dis)similarity (in terms of regular equivalence of units *a* and *b*) compared relative to their magnitude. REGE solves the problem that regular equivalence does not demand that two units that are equivalent be connected to the same number of equivalent units, by allowing that different links of *a* can be matched by the same link of *b*. It could be said that REGE algorithms (White, 2005) implicitly define regular equivalence for valued networks.

However, the two versions, REGGE (White, 1985a) and REGDI (White, 1985b) implicitly provide two different definitions of regular equivalence. The definition of REGGE (White, 1985a) would be that two units *a* and *b* are regularly equivalent, when for each link from *a* to *c* is matched by a link from *b* to *d*, where both values (of in- and outgoing arcs) of this link

[6] It should be noted that in REGE by White (1985a, b - both versions) a link represents a pair of arcs – an incoming and an outgoing arc.

are equal or greater to the values of the link from *a* to *c,* and *c* and *d* are regularly equivalent (in network $N = (U, R)$ and $a, b, c, d \in U$).

On the other hand, REGDI (White, 1985b) implies a more stringent definition of regular equivalence. The definition is very similar to that implied by REGGE (White, 1985a); only the condition "equal or greater" changes to "equal". The definition of REGDI (White, 1985b) would therefore be that two units *a* and *b* are regularly equivalent, when for each link from *a* to *c* is matched by a link from *b* to *d*, where both values (of in- and outgoing arcs) of this link are equal to the values of the link from *a* to *c*, and *c* and *d* are regularly equivalent (in network $N = (U, R)$ and $a, b, c, d \in U$). The same definition is implied by an older version of REGGE (White and Reitz, 1985).

It should be noted that REGE uses a slightly different interpretation of the definition of regular equivalence to the one is usually used in the generalized blockmodeling approach (and some other approaches) and in this text, as has been established. Nevertheless, the logic of the algorithm can be used, and even the algorithm itself can be easily adapted to the interpretation used in generalized blockmodeling. This was already noted by Borgatti and Everett (1993). This adaptation is presented in Section 5. This actually brings us very close to the ideas presented by Salier (1978: 79-82) in the first place, as is shown in Section 5. However, Salier did not define his structural relatedness as equivalence, and it does not possess (in general) symmetric and transitive properties.

Batagelj and Ferligoj (2000: 13) also presented some ideas for computing inconsistencies (generalized blockmodeling approach) for valued networks that can also be applied to regular equivalence. Based on their formula for computing inconsistencies for column-regular blocks, it can be seen that they define regular equivalence for valued networks in the following way. Two units are regularly equivalent if the maximums of the values of arcs from (and separately to) each of these two units to a group of equivalent units are the same for both units and each group of equivalent units separately.

The definition implied by Batagelj and Ferligoj (2000: 13) is very similar to the one implied by REGGE (White, 1985a). The only difference is in the understanding of the definition of regular equivalence, as was the case in binary networks (the difference lies in the separate versus joint treatment of possible arcs a → b and a ← b, where a and b are two vertices in the network). The generalization to valued networks is the same in both approaches.

f-regular equivalence

For valued networks, another type of equivalence, *f*-regular (or *function*-regular) equivalence, could be useful. This equivalence was suggested by Žiberna (2006b). That definition and description is repeated and extended in this subsection. This equivalence is not meant to be a strict generalization of regular equivalence, but to present a useful equivalence for valued networks that in some sense resembles regular equivalence. It tries to capture the idea that it is not necessary that equivalent units be equivalently connected to an individual unit, but only to a group of equivalent units. The idea of this set of possible equivalences is that a tie between a unit and a group of units can be adequately characterized by a function on the values of ties connecting this unit to the units of the selected group. This idea can be also recognized in the definition developed in the previous section.

Two units are *f*-regular equivalent if the values of some function *f* over the values of arcs from (and separately to) each of these two units to a group of equivalent actors are the same for both units and each group of equivalent units separately. If this function is *maximum*, we get the definition of the regular equivalence for valued networks implied by Batagelj and Ferligoj (2000: 13).

This definition can best be formally presented in matrix terms and is given below. The definition is given for single-relation networks, however it can be generalized to multi-relational networks by demanding that the definition hold for all relations.

Suppose that we have a network N = (U, R[7]) and ≡ is an equivalence relation on U that induces (or corresponds to) a partition **C** then ≡ is an *f-regular equivalence* (where f is a selected function, like *sum*, *maximum*, *mean*, etc.) if and only if for all $a, b \in U$ and all $X \in$ **C**, $a \equiv b$ implies:

(i) $\underset{i \in X}{f}\left(r_{ai}\right) = \underset{i \in X}{f}\left(r_{bi}\right)$ and

(ii) $\underset{i \in X}{f}\left(r_{ia}\right) = \underset{i \in X}{f}\left(r_{ib}\right)$.

This could be considered as another definition of the regular equivalence for valued networks or a new type of equivalence. The definition implied by Batagelj and Ferligoj (2000: 13) could be named *max*-regular and this one *sum*-regular, *mean*-regular or more generally *f*-regular, where f stands for a selected function. Which functions are suitable for specific analysis is not discussed here, although *sum* and *mean* at least (in addition to *maximum*) can be used in some circumstances. The choice the function mainly depends on how the strength of the tie between an individual and a group is (or should be) measured.

Different types[8] of *f*-regular equivalence can be seen as generalizations of different equivalences for binary data. This can be seen by applying these types of *f*-regular equivalence to binary data.

---

[7] The relation R is represented by matrix R = $[r_{ij}]_{n \times n}$

[8] In terms of f.

Max-regular equivalence is a generalization of regular equivalence. If the *maximums* of all rows and columns are 0, then the block is exactly null, while if they are all 1, then each row and each column has at least one tie and the block is obviously regular.

Sum-regular equivalence can be seen as a generalization of exact coloration as presented by Everett and Borgatti (1994: 43) to valued networks. The exact coloration states that equivalent vertices must have the same number of (in- and out-) neighbors of the same equivalence class. If each tie has the value 1, then this is exactly *sum*-regular equivalence.

However, if we change the definition of the *f*-regular equivalence to the following one and set *m* to 1, the definition of *sum*-regular and *max*-regular equivalence at level *m* complies with regular equivalence for binary networks.

Suppose that we have a network N = (*U*, *R*) and ≡ is an equivalence relation on *U* that induces (or corresponds to) a partition **C,** then ≡ is an *f-regular equivalence at level m* if and only if for all *a, b* ∈ *U* and all *X* ∈ **C**, *a* ≡ *b* implies both:

(i) $\underset{i\in X}{f}\left(r_{ai}\right)\geq m$ and $\underset{i\in X}{f}\left(r_{bi}\right)\geq m$ or $\underset{i\in X}{f}\left(r_{ai}\right)=\underset{i\in X}{f}\left(r_{bi}\right)=0$ and

(ii) $\underset{i\in X}{f}\left(r_{ia}\right)\geq m$ and $\underset{i\in X}{f}\left(r_{ib}\right)\geq m$ or $\underset{i\in X}{f}\left(r_{ia}\right)=\underset{i\in X}{f}\left(r_{ib}\right)=0$.

Generalized blockmodeling

Generalized blockmodeling was first introduced by Doreian, Batagelj, and Ferligoj (1994), where the notion of generalized equivalence and several block types were presented. The approach was described extensively by Doreian, Batagelj, and Ferligoj (2005), where descriptions of ideal blocks and formulas for computing the inconsistencies between empirical blocks and these ideal blocks were presented. However, it was still applicable only to binary and signed networks, although Batagelj and Ferligoj (2000) have presented some ideas for extending this approach to valued networks.

Žiberna (2006b) presented valued and homogeneity blockmodeling, two types of generalized blockmodeling suited to valued networks. He also compared the ideas presented by Batagelj and Ferligoj (2000) to his own suggestions and found them very similar to valued blockmodeling. These ideas formed the basis of implicit blockmodeling, which was, as already mentioned, first presented by Batagelj and Ferligoj (2000) and then further developed by Žiberna (2006b).

Valued and homogeneity blockmodeling were described in Žiberna (2006b), where definitions of *f*-regular and null (for valued blockmodeling only) ideal blocks and formulas for computing the inconsistencies between empirical blocks and these ideal blocks were also presented and discussed. Therefore this is not repeated here. Although Batagelj and Ferligoj (2000) suggested some formulas for computing inconsistencies using implicit blockmodeling for some ideal blocks (including null), regular blocks were not among them. Based on the work of Batagelj and Ferligoj (2000) and Žiberna (2006b), the formulas for computing inconsistencies for null and regular blocks for implicit blockmodeling are presented in Table 1. If the denominator in these formulas is zero, the block inconsistency is zero for null blocks and 1 for regular blocks. The description of ideal blocks to which these formulas correspond is the same as that for homogeneity blockmodeling (if function *maximum* is used as *f* in *f*-regular blocks) presented in Žiberna (2006b).

We could argue that null blocks are actually unnecessary. However, in this case, the inconsistency of the regular block should also be 0, when the denominator is 0.

The argument why the null blocks are unnecessary is the same as already provided by Žiberna (2006b) for homogeneity blockmodeling. The inconsistencies for (*max*-)regular blocks are appropriate for measuring both deviation from the ideal null blocks and deviation from the ideal (*max*-)regular block. This does not mean that when using only (*max*-)regular blocks, all blocks induced by the partition obtained using only these blocks should be considered (*max*-)regular. This should be determined using some other method. These inconsistencies measure if all row and column maximums are equal. If they are equal (or close) to 0, than this is essentially a null block.

For homogeneity blockmodeling, it is usually not appropriate to use both null and *max*-regular blocks together[9]. However, for implicit blockmodeling, the use of null and regular blocks is appropriate, while not necessary.

The formulas in Table 1 the nominator and the denominator play distinct roles. The nominator computes some measure of inconsistency, while the denominator normalizes it so that fraction always lies on the interval [0, 1]. However, the normalization is not necessary. If we use only the nominator, the inconsistencies for implicit blockmodeling become very similar to those for values blockmodeling (Žiberna, 2006b). The only difference that remains is that the parameter *m*, used in valued blockmodeling, is replaced by the maximum of the appropriate values in the implicit blockmodeling. In the case of null and regular blocks, these appropriate values are all values of a block. The non-normalized version of the implicit blockmodeling produced more reasonable results and was thus used in the empirical example in this paper.

---

[9] The symmetric nature of homogeneity approach makes the use of only *max*-regular blocks very suitable for computation of inconsistencies in null blocks. Usually, an inconsistency between an empirical block and *max*-regular ideal block would be smaller then an inconsistency between this empirical block and null ideal block, even if the empirical block would be very close to the ideal null block. In addition, there might be compatibility issues when using *max*-regular and null ideal blocks together.

Binary, valued and homogeneity blockmodeling were applied to an empirical example in Žiberna (2006b). In the empirical example section of this paper, only the final results of that application are presented. In addition, implicit blockmodeling (non-normalized version) is also applied to that example and compared with the previous results.

REGE

There exists no "closed form[10]" measure of dissimilarity or similarity compatible with regular equivalence. However, there does exist an algorithm that produces in iterations a sequence of measures of dissimilarity or similarity (depending on the version) in terms of regular equivalence (White, 2005). For blockmodeling purposes, this (dis)similarity matrix must be analyzed using an appropriate clustering algorithm. Hierarchical clustering algorithms have usually been used for this purpose. For example, Ucinet 5 (Borgatti, Everett, and Freeman, 1999) uses single linkage hierarchical clustering to find an appropriate partition. However, based on my experience, other clustering algorithms, such as Ward's method and complete linkage, perform better.

The algorithm (REGE) was described in the section on regular equivalence, or more precisely, in the subsection on valued networks. This description is therefore not repeated here. However, as has been noted by Borgatti and Everett (1993:385) "…there are certain aspects of REGE's operation that are not altogether clear".

A few descriptions of the algorithm can be found in the literature; however, they are not sufficient. The key paper by White and Reitz (1985) remained unpublished[11]. Additionally it

[10] No measure that could be computed in closed form exists.

[11] To the best of my knowledge, the first published paper containing the formulas for computing REGE similarities was written by Faust (1988, 325-328).

does not describe the algorithm (White, 1985a) that can be found on Douglas R. White's [12] web page (White, 2005)[13], while the description by Borgatti and Everett (1993) focuses on binary networks and does not indicate how REGE handles the values of ties. Some additional papers have been written (White,1984a; White, 1980; 1982; and 1984b in Borgatti and Everett, 1993) on this topic. Unfortunately they have also remained unpublished.

In this section the algorithm and some of its variants[14] are described mainly by means of formulas used by the algorithms to compute similarities or dissimilarities in terms of regular equivalence. It is hoped that the formulas and the accompanying descriptions are sufficient for an understanding of the algorithm and the difference between the several existing variants.

Existing versions (or descriptions) of REGE

Notation

${}_{ij}M_{km}$ … Matches of ties between unit i and unit k to ties between unit j and unit k

${}_{ij}Max_{km}$ … Maximum possible matches of ties between unit i and unit k to ties between unit j and unit k

t … iteration (number)

E_{ij}^{t} … Degree of equivalence between unit i and unit j in iteration t (similarity or dissimilarity matrix).

---

[12] The author of the REGE algorithm and coauthor of the original definition of regular equivalence.

[13] Based on the comparison of the results obtained, I assume that the same algorithm is used in Ucinet 5 (Borgatti et. al, 1999).

[14] Only the variants aimed at regular equivalence and data measured on at least an interval scale are considered here.

B … A binary relation based on R represented by binary matrix B with cells b_{ij}, where $b_{ij} = 1$, $r_{ij} > 0$ and 0 otherwise.

S … A symmetric binary relation based on R represented by binary matrix S with cells s_{ij}, where $s_{ij} = 1$, if $r_{ij} + r_{ij} > 0$ and 0 otherwise.

$\overset{*}{\max\limits_{m(ijkm)}}$ indicated that in the denominator, an element that corresponds to the maximum in the numerator is selected. The optional indices in brackets indicate to which of the maximums in the numerator the element must correspond.

A version of the REGE algorithm described in White and Reitz (1985, 14-15,17-18) is presented in the following equations, limited to a network with only one relation (for reasons of simplicity). White and Reitz (1985, 14-20) presented several measures[15] of regular equivalence. Here, only the one which can be also found in a published paper written by Faust (1988, 325-328) is presented, since the others were never officially published.

REGGE-O(Old): REGE algorithm from White and Reitz (1985, 18)

$$_{ij}M_{km} = \min(r_{ik}, r_{jm}) + \min(r_{ki}, r_{mj})$$
$$_{ij}Max_{km} = \max(r_{ik}, r_{jm}) + \max(r_{ki}, r_{mj})$$
$$\mathbf{E}_{ij}^{t+1} = \frac{\sum_{k=1}^{n} \max_{m=1}^{n} \left(\mathbf{E}_{km}^{t} \left(_{ij}M_{km}^{t} + _{ji}M_{km}^{t} \right) \right)}{\sum_{k=1}^{n} \overset{*}{\max_{m}} \left(_{ij}Max_{km} + _{ji}Max_{km} \right)}$$

---

[15] White and Reitz (1985, 14-20) present two main version of the algorithm presented below that differ only in the case of multi-relational networks. Since only single-relation networks are addressed here, this is not relevant. Additionally, they present two starting options for an initial similarity matrix (E^0), of which only one is presented here. In addition, they present a similar set of algorithms which compute regular dissimilarities instead of regular similarities. These are also not covered here (since they were never officially published); however a newer version is discussed (see REGDI algorithm below).

The first two equations have to be computed only once, since they do not include any reference to t, the iteration number. The third equation presents the computation of similarity matrix (E^{t+1}) in iteration $t + 1$ based on the similarity matrix (E^t) in the previous iteration and the quantities computed in the above formulas. For the first iteration, a matrix filled with 1s is used as a similarity matrix from the previous iteration (E^0). The formulas, however, have a strange feature. As we can see, in the first two formulas, the quantities ${}_{ij}M_{km}$ and ${}_{ij}Max_{km}$ have no reference to the iteration number (t). However, in the third formula, it has an index t, which indicates the iteration number. This is probably left over from an even earlier version of the algorithm, where ${}_{ij}M_{km}$ and ${}_{ij}Max_{km}$ did depend on the iteration (White, 1984a).

Another inconsistency with both the definition of regular equivalence (White and Reitz, 1983: 200) and the description of REGE (plus examples) in Borgatti and Everett (1993: 364-369) can be found. Firstly, in the numerator, the maximum should be computed separately for ${}_{ij}M_{km}$ and separately for ${}_{ji}M_{km}$; however, in this formula it is computed for ${}_{ij}M_{km} + {}_{ji}M_{km}$. Similarly, the corresponding ${}_{ij}Max_{km}$ should be chosen in the denominator. The denominator is also computed incorrectly, since ${}_{ij}Max_{km}$ should only be used in the sum, if $r_{ik} + r_{ki}$ is greater than 0 (only matches for links are searched for, and 0 indicates the absence of a link).

If we correct all these inconsistencies, we get the formulas written below. It can be noted that the formulas for ${}_{ij}M_{km}$ and ${}_{ij}Max_{km}$ do not change.

The recursive formula changes in such a way that the maximums for ${}_{ij}M_{km}$ and ${}_{ji}M_{km}$ are now computed separately and the corresponding ${}_{ij}Max_{km}$ is likewise chosen. However, one detail still has to be set. This is which ${}_{ij}Max_{km}$ is chosen, when there are several maximal ${}_{ij}M_{km}$. In that case, we select the minimal ${}_{ij}Max_{km}$ out of those that correspond to the maximal ${}_{ij}M_{km}$. This way, if an exact match to r_{ik} exists, it will be selected.

REGGE-OC(Old Corrected): Corrected REGGE-O algorithm

$$
{}_{ij}M_{km} = \min(r_{ik}, r_{jm}) + \min(r_{ki}, r_{mj})
$$

$$
{}_{ij}Max_{km} = \max(r_{ik}, r_{jm}) + \max(r_{ki}, r_{mj})
$$

$$
\mathbf{E}_{ij}^{t+1} = \frac{\sum_{k=1}^{n} \left(\max_{m=1}^{n} \left(\mathbf{E}_{km}^{t}\, {}_{ij}M_{km} \right) + \max_{m=1}^{n} \left(\mathbf{E}_{km}^{t}\, {}_{ji}M_{km} \right) \right)}{\sum_{k=1}^{n} \left(s_{ik} \max_{m(ij)}^{*} \left({}_{ij}Max_{km} \right) + s_{jk} \max_{m(ji)}^{*} \left({}_{ji}Max_{km} \right) \right)}
$$

This version would produce the same exact regular equivalence classes as the algorithm REGDI presented later.

However, this is still not the formula used in REGGE (White, 1985a). That formula uses a simpler denominator and is presented in the formulas below. The simplification of the denominator means that an tie with any value greater than or equal to the value of the tie for which a match is searched perfectly matches this tie. The contrary does not hold. This was not the case in the above formulas, where for a perfect match, the ties had to have identical values. The same is also true for REGDI, an algorithm presented later.

REGGE: REGE algorithm as implemented in REGGE (White, 1985a)[16]:

$$
{}_{ij}M_{km} = \min(r_{ik}, r_{jm}) + \min(r_{ki}, r_{mj})
$$

$$
\mathbf{E}_{ij}^{t+1} = \frac{\sum_{k=1}^{n} \left(\max_{m=1}^{n} \left(E_{km}^{t}\, {}_{ij}M_{km} \right) + \max_{m=1}^{n} \left(\mathbf{E}_{km}^{t}\, {}_{ji}M_{km} \right) \right)}{\sum_{k=1}^{n} \left(r_{ik} + r_{ki} + r_{jk} + r_{kj} \right)}
$$

White(1985b) also developed a distance version of REGE, which he called REGDI.

REGDI: REGE algorithm as implemented in REGDI (White, 1985b):

[16] The formula was confirmed by the author of the original REGE algorithm, Douglas R. White, through personal communication. Douglas R. White also helped the author of this paper get a better understanding of the REGGE algorithm.

$$ {}_{ij}M_{km} = \left(r_{ik}\text{-}r_{jm}\right)^2 + \left(r_{ki}\text{-}r_{mj}\right)^2 $$

$$ \mathbf{E}_{ij}^{t+1} = \min\left(\frac{\sum_{k=1}^{n}\left(\min_{m=1}^{n}\left(\max\left({}_{ij}M_{km}, \mathbf{E}_{km}^{t}\left(r_{ik}^2 + r_{ki}^2\right)\right)\right) + \min_{m=1}^{n}\left(\max\left({}_{ji}M_{km}, \mathbf{E}_{km}^{t}\left(r_{jk}^2 + r_{kj}^2\right)\right)\right)\right)}{\sum_{k=1}^{n}\left(r_{ik}^2 + r_{ki}^2 + r_{jk}^2 + r_{kj}^2\right)}, 1 \right) $$

In these formulas, E^t is a dissimilarity matrix for *t*-th iteration. The computations here are not as clear as in the previous cases. If we look closely at the formulas, we can see that in the evaluation of each pair of links either only equivalence (through dissimilarity in terms of regular equivalence) of the units at the other end (not those currently evaluated for regular equivalence) of the link or only the difference between values of ties is used in the contribution of the link in question to the dissimilarity in terms of regular equivalence.

However, the numerator can not simply be

$$ \sum_{k=1}^{n}\left(\min_{m=1}^{n}\left({}_{ij}M_{km}\mathbf{E}_{km}^{t}\right) + \min_{m=1}^{n}\left({}_{ji}M_{km}\mathbf{E}_{km}^{t}\right)\right) $$

since this quantity $\min_{m=1}^{n}\left({}_{ij}M_{km}\mathbf{E}_{km}^{t}\right)$ would always be 0, since $\mathbf{E}_{km}^{t}$ for $k = m$ is always zero. The algorithm used is an attempt to solve this problem. It forces the algorithm to focus on differences, when the units at the other end of links are quite similar (in the sense of regular equivalence) and to focus on the dissimilarities in the sense of regular equivalence, when the differences in values of ties are relatively small.

Suggested modifications to REGE

As has been established, REGE does not fully comply with the interpretation of the definition of regular equivalence used in generalized blockmodeling. In generalized blockmodeling, the arcs r_{ij} and r_{ji} are treated independently.

Only a slight modification is needed to modify the REGE algorithm(s) to have this property. The modified algorithms are presented below. They are named "One Way" (OW), since here each (arc) *way* of the dyad is evaluated independently, "one way" at the time.

REGGE-OCOW(One Way): One way version of the REGGE-OC

$$\mathbf{E}_{ij}^{t+1} = \frac{\sum_{k=1}^{n} \begin{pmatrix} \max_{m=1}^{n}\left(\mathbf{E}_{km}^{t} \min\left(r_{ik}, r_{jm}\right)\right) + \max_{m=1}^{n}\left(\mathbf{E}_{km}^{t} \min\left(r_{ki}, r_{mj}\right)\right) + \\ + \max_{m=1}^{n}\left(\mathbf{E}_{km}^{t} \min\left(r_{jk}, r_{im}\right)\right) + \max_{m=1}^{n}\left(\mathbf{E}_{km}^{t} \min\left(r_{kj}, r_{mi}\right)\right) \end{pmatrix}}{\sum_{k=1}^{n} \begin{pmatrix} \max_{m(ijkm)}^{*}\left(b_{ik} \max\left(r_{ik}, r_{jm}\right)\right) + \max_{m(kmij)}^{*}\left(b_{ki} \max\left(r_{ki}, r_{mj}\right)\right) + \\ + \max_{m(jikm)}^{*}\left(b_{jk} \max\left(r_{jk}, r_{im}\right)\right) + \max_{m(kmji)}^{*}\left(b_{kj} \max\left(r_{kj}, r_{mi}\right)\right) \end{pmatrix}}$$

The only modification used here is that we no longer search for the maximum of ${}_{ij}M_{km} = \min\left(r_{ik}, r_{jm}\right) + \min\left(r_{ki}, r_{mj}\right)$, but separately for the maximum of $\min\left(r_{ik}, r_{jm}\right)$ and separately of $\min\left(r_{ki}, r_{mj}\right)$. Similarly, the suitable maximums are chosen separately.

REGGE-OW (One Way): One way version of the REGGE algorithm

$$\mathbf{E}_{ij}^{t+1} = \frac{\sum_{k=1}^{n} \begin{pmatrix} \max_{m=1}^{n}\left(\mathbf{E}_{km}^{t} \min\left(r_{ik}, r_{jm}\right)\right) + \max_{m=1}^{n}\left(\mathbf{E}_{km}^{t} \min\left(r_{ki}, r_{mj}\right)\right) + \\ + \max_{m=1}^{n}\left(\mathbf{E}_{km}^{t} \min\left(r_{kk}, r_{im}\right)\right) + \max_{m=1}^{n}\left(\mathbf{E}_{km}^{t} \min\left(r_{kj}, r_{mi}\right)\right) \end{pmatrix}}{\sum_{k=1}^{n}\left(r_{ik} + r_{ki} + r_{jk} + r_{kj}\right)}$$

In this version of REGE (REGGE-OW) the definition of regular equivalence for valued networks used is the same as implied by the implicit approach of Batagelj and Ferligoj (2000: 13), which was called *max*-regular equivalence (in the equivalence section of this paper).

This algorithm is very close to a symmetric version[17] of one of the measures of structural relatedness suggested by Salier (1978: 79-82), where both in- and outgoing ties are taken into

[17] Salier (1978) did not propose a symmetric version.

account. For use in this algorithm, which is presented below, the matrix R should be normalizes beforehand to the interval [0,1].

An algorithm for computing similarities in terms of structural relatedness as proposed by Salier (1978: 79-82)[18]:

$$\mathbf{E}_{ij}^{t+1} = \frac{\sum_{k=1}^{n}\left(\max_{m=1}^{n}\left(\min\left(\mathbf{E}_{km}^{t}, r_{ik}, r_{jm}\right)\right) + \max_{m=1}^{n}\left(\min\left(\mathbf{E}_{km}^{t}, r_{ki}, r_{mj}\right)\right)\right)}{\sum_{k=1}^{n}\left(r_{ik} + r_{ki}\right)}$$

A symmetric version of Salier's algorithm:

$$\mathbf{E}_{ij}^{t+1} = \frac{\sum_{k=1}^{n}\left(\begin{array}{l}\max_{m=1}^{n}\left(\min\left(\mathbf{E}_{km}^{t}, r_{ik}, r_{jm}\right)\right) + \max_{m=1}^{n}\left(\min\left(\mathbf{E}_{km}^{t}, r_{ki}, r_{mj}\right)\right) + \\ + \max_{m=1}^{n}\left(\min\left(\mathbf{E}_{km}^{t}, r_{jk}, r_{im}\right)\right) + \max_{m=1}^{n}\left(\min\left(\mathbf{E}_{km}^{t}, r_{kj}, r_{mi}\right)\right)\end{array}\right)}{\sum_{k=1}^{n}\left(r_{ik} + r_{ki} + r_{jk} + r_{kj}\right)}$$

REGDI-OW (One Way): One way version of the REGDI algorithm

$$\mathbf{E}_{ij}^{t+1} = \min\left(\frac{\sum_{k=1}^{n}\left(\begin{array}{l}\min_{m=1}^{n}\left(\max\left(\left(r_{ik} - r_{jm}\right)^2, \mathbf{E}_{km}^{t} r_{ik}^2\right)\right) + \min_{m=1}^{n}\left(\max\left(\left(r_{ki} - r_{mj}\right)^2, \mathbf{E}_{km}^{t} r_{ki}^2\right)\right) + \\ + \min_{m=1}^{n}\left(\max\left(\left(r_{jk} - r_{im}\right)^2, \mathbf{E}_{km}^{t} r_{jk}^2\right)\right) + \min_{m=1}^{n}\left(\max\left(\left(r_{kj} - r_{mi}\right)^2, \mathbf{E}_{km}^{t} r_{kj}^2\right)\right)\end{array}\right)}{\sum_{k=1}^{n}\left(r_{ik}^2 + r_{ki}^2 + r_{jk}^2 + r_{kj}^2\right)}, 1\right)$$

Artificial examples

In an attempt to clarify the differences among different REGE algorithms, several artificial networks were constructed and analyzed using REGE. This is not necessary for direct

[18] One of the measures of structural relatedness suggested by Salier (1978: 79-82), where both in- and outgoing ties are taken into account.

approaches, as they have clearly defined ideal blocks. In addition, binary and valued blockmodeling are not suited to such examples, where we cannot select one slicing parameter or one value of the parameter *m* that is suitable for all blocks. Implicit and homogeneity blockmodeling (with *max*-regular equivalence) are very suitable for these kinds of examples; however, in these idealized settings they would always produce the same partitions as REGGE-OW. The exception occurs if we seek a regular equivalence partition that does not correspond to the maximal regular equivalence. All versions of REGE are only suitable for finding (an approximation to) maximal regular equivalence partitions, while generalized blockmodeling approaches can also find other regular equivalence partitions.

The network in Figure 1 is a very simple network. It is a classical core-periphery model. The network has a clear partition into core units (1-4) and periphery units (5-10). The network can be considered undirected, since each arc has a reciprocal arc. The matrix representing it is therefore symmetric. If this was a binary network (with all non-null values having a value of 1), all versions of REGE described above would find that all units together form one regular equivalence class. However, in the valued case, all versions produce two classes that can be seen in Figure 1. In the matrix representation, the classes are separated by a thicker black line, while in the graph representation, they are distinguished by vertex colors (white – core, black – periphery).

Network 1 was symmetric. Network 2 is very similar; however, in the ties among the core (1-4) units and units 5-7, this symmetry is broken. The periphery class is split into two classes, as indicated by the additional gray[19] line on the matrix in Figure 2. The difference between these two classes is best seen in the graph in Figure 2. The black vertices are unchanged and have both an incoming and an outgoing arc with at least one of the white vertices. The gray

[19] The gray line indicates a split, found by only some versions of REGE.

vertices are also joined with the white vertices through incoming and outgoing arcs; however, no gray vertex has both an incoming and an outgoing arc with the same white vertex.

Here the difference between algorithms where symmetry matters and those where it does not (all "one way" algorithms, marked with OW in the previous section) comes into focus. The "one way" (or OW) algorithms find the same two regular equivalence classes as on network 1 (only the split indicated by the black line on the matrix in Figure 2), while the other approaches find 3 regular equivalence classes presented in Figure 2 (the additional split is indicated by the gray line).

The second difference among the regular equivalence classes that these algorithms seek corresponds to what kind of values in a block have to be equal in order for the block to be considered regular. For REGGE (when the symmetry requirement mentioned above is satisfied) and REGGE-OW algorithms, only row and column maximums have to be equal. On the other hand, for the other 4 versions of REGE, each row and each column have to contain the same values (ignoring zeros), where each unique value can appear one or more times. The network 1, modified so this effect can be shown (network 3), is presented in Figure 3. In this network the REGGE and REGGE-OW algorithms find the same two classes as are shown in Figure 1 and are unaffected by the addition of the arcs with value 2 to the blocks *R(2,1)* and *R(1,2)*.

The other algorithms find three regular equivalence classes presented in Figure 3. The black vertices are the same as they were in Figure 1, while the gray vertices have additional two-way arcs with value 2 linking them to the white vertices.

The same difference in the characteristics of the REGGE, REGGE-OC and REGDI algorithms can also be shown for network 4 in Figure 4, although the blocks *R(2,1), R(3,1), R(1,2)* and *R(1,3)* still contain only arcs with value 5. However, in network 4, block *R(2,1)* contains the arcs from network 1 and network 2 (from the same block). Each gray vertex has now at least one unreciprocal arc from a white vertex and at least one reciprocal tie with a white vertex. One versus two arcs in a dyad now takes the place of different values. The REGGE algorithm finds only two regular equivalence classes in this network, while the REGDI and REGGE-OC algorithms find three regular equivalence classes shown in Figure 4.

Empirical example: Note-borrowing between social-informatics students (line measurement)

The data in this example come from a survey conducted in May 1993 on 13 social-informatics students (Hlebec, 1996: 90). The network was constructed from answers[20] to the question, "How often did you borrow notes from this person?" for each of the fellow students. The results are presented in Figure 5 in matrix and graph format.

Generalized blockmodeling

The aim of the analysis is to discover groups of students that play similar roles. Only groups in terms of (*max*-)regular equivalence were sought. This network was analyzed by Žiberna (2006b) using binary, valued and homogeneity blockmodeling. The best partitions found by Žiberna (2006b) using these approaches are presented in Figure 6 (one for each approach). Based on Žiberna (2006b), only 3 cluster partitions were considered. Where the approach produces an image (binary, valued and later implicit blockmodeling with null and regular blocks), the null blocks are indicated on the matrix by the shades of lines instead of shades of the solid color indicating the tie strength. These shaded lines are hard to see when tie values are low; however, this itself indicates null blocks.

[20] The respondents indicated the frequency of borrowing by choosing (on a computer) a line of length 1 to 20, where 1 meant no borrowing. 1 was deducted from all answers, so that 0 now means no borrowing.

Binary blockmodeling works reasonably well. The partition would, however, be much better if the unit 8 was in the third cluster. However, this partition does allow a reasonable interpretation. We have a primary core (units 3,4, and 9) that lend notes to all students, a second group that could represent the main body of students, who borrow notes from everybody, but primarily from the core, and a peripheral group of students (1,7, and 8) that borrow notes from the core and lend them to the main body of students.

Valued blockmodeling produced what is probably the best partition of all approaches. The third group of students lends notes to everybody; the second borrows notes from the third group and also shares them within (the group), while the first group only borrows notes from the third group.

The partitions obtained using homogeneity (sum of squares and absolute deviations) blockmodeling do not seem as clear as valued blockmodeling. Both homogeneity approaches were good at finding the two or three most important "suppliers" of notes, while unit 3 in valued blockmodeling partition does not seem to fit the third group. However, the partition of the remaining units is less satisfactory. The partition obtained with absolute deviations seems better then the sum of squares partition. The homogeneity blockmodeling partitions also suggest a different interpretation. In this case, there is more "traffic" between the first and the second group than within these groups. Students of the second group in particular quite often borrow notes from students of the first group. That may be an indication that these groups represent two groups of students that usually attend the same classes, or it may just show that the partition is not appropriate.

Implicit blockmodeling was not applied to this example by Žiberna (2006b), since the algorithm has not yet been implemented in the package `blockmodeling` at that time. Therefore, the analysis is presented here. Two results are presented Figure 7, one with allowed ideal blocks null and regular and one with regular ideal blocks only.

The partition obtained using null and regular blocks is acceptable; however only one block is classified as regular. The partition obtained using only regular blocks is even better. It is similar to the absolute deviations partition; however blocks *R(1,2), R(2,2)* and *R(3,1)* appear cleaner. On the other hand, the additional member of the third group does not exactly fit. The interpretation nevertheless stays the same.

REGE

All versions of REGE presented in this paper were also applied to this example. The algorithms run for 10 interactions, so that the differences between subsequent iterations were small. Since the output of all versions is a (dis)similarity matrix, a clustering algorithm is needed to find a partition. The choice of an appropriate clustering algorithm is essential. On the similarity matrix obtained with REGGE, three hierarchical clustering algorithms (Ward's method, complete linkage, and single linkage) were tested to demonstrate the effect of a clustering algorithm. Their dendrograms are presented in Figure 8. Ward's method and complete linkage produce the same partition at three (and four) clusters, while the three cluster single linkage partition is profoundly different and does not partition the matrix well. Ward's method and complete linkage produced the same partitions in about half the cases; however only partitions produces by Ward's method are presented (since they produced superior results).

The first matrix in Figure 9 presents the partition produced by REGGE, REGGE-OW, and REGDI-OW using Ward's or complete linkage hierarchical clustering.

The other three algorithms gave different partitions. The partitions obtained using these algorithms also differed depending on the hierarchical clustering algorithm used (Ward's or complete linkage, only partitions obtained using Ward's method are presented). These partitions are presented on the remaining three matrices in Figure 9.

The best partition out of all REGE partitions seems to be the one obtained using REGGE-OC. It is very similar to the absolute deviations and implicit blockmodeling (with only regular blocks) partitions, and it also offers a similar interpretation. Other partitions seem to be quite good at finding the main lender group, but not as good at partitioning the other.

However, the results of this analysis should not be used to compare these algorithms, as only one network is not sufficient for this purpose. The section on REGE and the previous section covering several artificial examples should be consulted regarding what kind of ideal partitions are produced or found by the REGE algorithms. For the comparison of these algorithms on non-ideal networks, a simulation study or a comparison involving several real networks should be done.

Discussion

The paper shows that regular equivalence is not ambiguously defined for valued networks. Several versions of the definition exist. It was also shown that different algorithms developed for detecting and measuring regular equivalence for valued and binary networks assume different definitions of regular equivalence.

An alternative equivalence for valued networks, an *f*-regular equivalence (Žiberna, 2006b), was also discussed. Generalized blockmodeling of valued networks can be used to detect and measure *f*-regular equivalence. This approach includes two approaches, valued and homogeneity blockmodeling. Implicit blockmodeling, an approach suggested by Batagelj and Ferligoj (2000) and further discussed and developed by Žiberna (2006b) and in this paper, could also be seen as a generalized blockmodeling (of valued networks) approach. It can be used to detect and measure *max*-regular equivalence (a version of *f*-regular equivalence).

Other approaches capable of detecting and/or measuring regular equivalence in interval valued networks were also reviewed. These approaches are binary generalized blockmodeling (Doreian, Batagelj, and Ferligoj, 2005) and three versions of the REGE algorithm (White, 2005; White and Reitz, 1985; Faust, 1988). Some modifications to the REGE algorithms were also suggested and implemented. All these approaches were applied to an empirical example using the R package `blockmodeling` (Žiberna, 2006a).

From the descriptions of the algorithms and approaches that have been evaluated and also from the application of the REGE algorithms to a few artificial examples, it can be seen that in ideal cases (in case of perfect correspondence of a selected network to an appropriate definition of regular equivalence), the following groups or pairs of approaches produce the same partitions:

- REGGE-OC and REGDI,
- REGGE-OCOW and REGDI-OW, and
- REGGE-OW[21], implicit blockmodeling, and homogeneity blockmodeling with *max*-regular equivalence.

However, application of these approaches showed that on empirical networks, these algorithms can produce different partitions. In the example, all the approaches that were compared revealed the same basic network structure: a group (the third group) that is a main provider of notes and two other groups. However, different approaches show different patterns of exchange within and between these two groups. The partition that seems the most reasonable is obtained using valued blockmodeling. In this partition, the majority of the remaining exchange of notes takes place within the second group. A similar partition is found using binary blockmodeling. On the other hand, in homogeneity and implicit blockmodeling

[21] REGGE-OW can only be used to find maximal regular equivalence partitions, while generalized blockmodeling approaches can also find other regular partitions.

partitions, most of this remaining exchange takes place between the remaining two groups. Most other partitions fall somewhere between these.

The advantage of the indirect approaches (all versions of REGE) is their speed, and therefore they are also the only approaches feasible on networks with more than 255 (current limitation for generalized blockmodeling in Pajek (Batagelj and Mrvar, 2006)) vertices. However, they do not directly produce the partition, only a matrix of (dis)similarities. Even to obtain this matrix, the number of iterations must first be chosen, for which only "a rule of thumb" directive exists. After this matrix of (dis)similarities is obtained, a suitable clustering algorithm must be selected and used, thus requiring some knowledge of cluster analysis. Although it has been established what kind of ideal partitions they would find and which definitions of regular equivalence they use, they do not have a clearly defined criterion function that they optimize. This is a disadvantage from the blockmodeling perspective. However it does have other favorable consequences. The (dis)similarity matrix thus obtained can be used for other purposes, such as multidimensional scaling and fuzzy clustering. This might be more appropriate in some cases, since crisp partitions do not always appear in real life situations. Another disadvantage of REGE algorithms is that they can only be used to find maximal regular partitions. This is especially undesirable if the maximal regular partition is trivial (if it is composed of only one cluster). Fortunately, such cases practically do not exist in valued networks.

On the other hand, the direct approaches directly produce a partition and a measure of how inconsistent this partition is (on the network analyzed) with an ideal (*f*-)regular equivalence[22]. No additional steps are required; however the number of clusters must be chosen in advance. This measure is a clearly defined criterion function that they minimize. Additionally, the direct approaches can be used to obtain an image that corresponds to a partition and also to

[22] The definition of the ideal (*f*-)regular equivalence and of this measure depends on the approach used.

find a partition that best corresponds to a certain image (for regular equivalence, this means the position of regular and null blocks in the image). This is especially true for binary and valued blockmodeling[23]. Although when using binary or valued blockmodeling a parameter must be selected in advance, its value can be approximately determined based on network characteristics and knowledge of the subject matter. Actually, this parameter is needed to completely define the modified version of (*f*-)regular equivalence used by these approaches.

References

Batagelj, V., Doreian, P., & Ferligoj, A. (1992). An optimizational approach to regular equivalence. *Social Networks, 14,* 121–135.

Batagelj, V., & Ferligoj, A. (2000): Clustering relational data. In W. Gaul, O. Opitz, & M. Schader (Eds.), *Data Analysis* (pp. 3 – 16). New York: Springer-Verlag.

Batagelj, V., & Mrvar, A. (2006). *Pajek 1.11*, Retrived January 6, 2006, from http://vlado.fmf.uni-lj.si/pub/networks/pajek/.

Borgatti, S.P., & Everett, M.G. (1992). Regular Blockmodels of Multiway, Multimode Matrices. *Social Networks, 14*, 91-120.

Borgatti, S.P., & Everett, M.G. (1993). Two algorithms for computing regular equivalence. *Social Networks, 15*, 361-367.

Borgatti, S.P., Everett, M.G., & Freeman, L.C. (1999). *Ucinet 5 for Windows: Software for Social Network Analysis*. Natic: Analytic Technologies.

[23] Although this can not be seen from this paper, homogeneity blockmodeling can be also used for this purpose, if (and only if) a fixed (for a given block) central value (as an alternative to 0 in null blocks) can be selected for *f*-regular blocks.

Doreian, P., Batagelj, V., & Ferligoj, A. (2005). *Generalized Blockmodeling*. New York: Cambridge University Press.

Everett, M.G. & Borgatti, S.P. (1994). Regular equivalence: General theory. *Journal of Mathematical Sociology, 19*, 29-52.

Faust, K. (1988). Comparison of methods for positional analysis: Structural and general equivalences. *Social Networks, 10*, 313–341.

Hlebec, V. (1996). *Metodološke značilnosti anketnega zbiranja podatkov v analizi omrežji: Magistersko delo*. Ljubljana : FDV.

R Development Core Team (2005). *R: A language and environment for statistical computing*. Vienna: R Foundation for Statistical Computing. Retrieved May 12, 2005 from http://www.R-project.org

Sailer, Lee Douglas (1978): Structural equivalence: Meaning and definition, computation and application. *Social Networks, 1*, 73-90.

Smith, D. A., & White, D.R. (1992). Structure and Dynamics of the Global Economy: Network Analysis of International Trade 1965-1980. *Social Forces*, *70*, 856-893.

White, D.R. (1985a). *DOUG WHITE'S REGULAR EQUIVALENCE PROGRAM*. Retrieved May 12, 2005 from http://eclectic.ss.uci.edu/~drwhite/REGGE/REGGE.FOR.

White, D.R. (1985b). *DOUG WHITE'S REGULAR DISTANCES PROGRAM*. Retrieved May 12, 2005 from http://eclectic.ss.uci.edu/~drwhite/REGGE/REGDI.FOR.

White, D.R. (2005). *REGGE (web page)*. Retrieved May 12, 2005 from http://eclectic.ss.uci.edu/~drwhite/REGGE/.

White, D.R. (1980). Structural equivalences in social networks: concepts and measurement of role structures. *Paper presented at Research Methods in Social Network Analysis Conference, Laguna Beach, California, April 1980*.

White, D.R. (1982). Measures of global role equivalence in social networks. *Unpublished manuscript.*

White, D.R. (1984a). Measuring Role Distance: An Algorithm for REGular Graph Equivalence (REGGE). *Unpublished manuscript.*

White, D.R. (1984b). REGGE: a REGular Graph Equivalence algorithm for computing role distances prior to blockmodeling. *Unpublished manuscript.*

White, D. R., & Reitz, K. P. (1983). Graph and semigroup homomorphisms on networks of relations. *Social Networks, 5*, 193-234.

White, D. R., & Reitz, K. P. (1985). Measuring role distances: Structural, regular and relational equivalences. *Unpublished manuscript*. University of California, Irvine.

Žiberna, A. (2006a). *blockmodeling 0.1.2: An R package for Generalized and classical blockmodeling of valued networks*. Retrieved March 15, 2006 from http://www2.arnes.si/~aziber4/blockmodeling/.

Žiberna, A. (2006b). Generalized Blockmodeling of Valued Networks. *Social Networks (in press)* Retrieved May 30, 2006 from http://dx.doi.org/10.1016/j.socnet.2006.04.002.

Tables

Table 1: Characterizations of ideal blocks[24]

Ideal block	Description blockmodeling[25]	Block inconsistencies - $\delta(R(C_a, C_b), T)$	Position of the block
null	all 0 (an exception may be cells on the diagonal, where all values should then be equal)	$\frac{\sum_{i=1}^{n_r}\sum_{j=1}^{n_c} \mathrm{r}_{ij}}{n_r n_c \max_{i,j} r_{ij}}$	*off-diagonal*
		$\frac{\sum_{i=1}^{n_r}\sum_{j=1}^{n_c} \mathrm{r}_{ij} - \min\left(0,2\sum_{i=1}^{n_r} r_{ii} - n_r \sum_{i=1}^{n_r} \max_{\mathrm{i,j}} r_{ij\,ii}\right)}{n_r n_c \max_{i,j} r_{ij}}$	*diagonal*
regular	maximum over all rows and all columns equal	$\frac{\sum_{i=1}^{n_r}\sum_{j=1}^{n_c} \max\left(\left(\max_{i,j} r_{ij} - \max_{j} r_{ij}\right),\left(\max_{i,j} r_{ij} - \max_{i} r_{ij}\right)\right)}{n_r n_c \max_{i,j} r_{ij}}$	

[24] Since this paper focuses on regular equivalence, some ideal blocks are omitted. Their descriptions can be found in Žiberna (2006b).

[25] Slightly adapted from Doreian et. al, 2005: 224.

Figures

Figure 1: Network 1

	1	2	3	4	5	6	7	8	9	10
1	10	10	10	10	5		5		5	
2	10	10	10	10		5				5
3	10	10	10	10			5		5	
4	10	10	10	10	5			5		
5	5			5						
6		5								
7	5		5							
8				5						
9	5		5							
10		5								

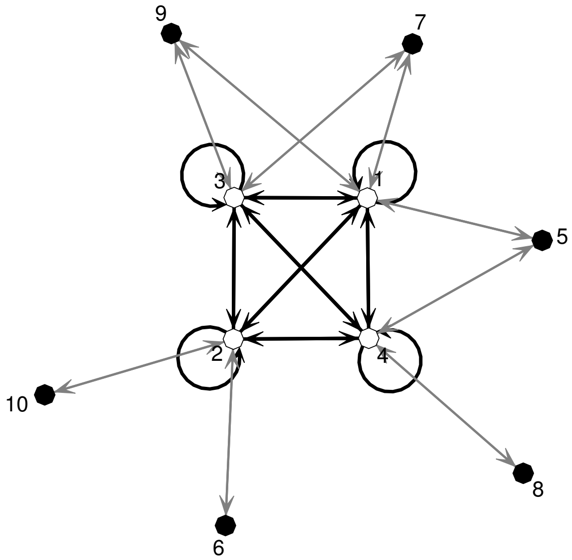

Figure 2: Network 2

	1	2	3	4	8	9	10	5	6	7
1	10	10	10	10		5			5	
2	10	10	10	10			5	5		5
3	10	10	10	10		5		5		
4	10	10	10	10	5					5
8				5						
9	5		5							
10		5								
5	5			5						
6		5								
7	5		5							

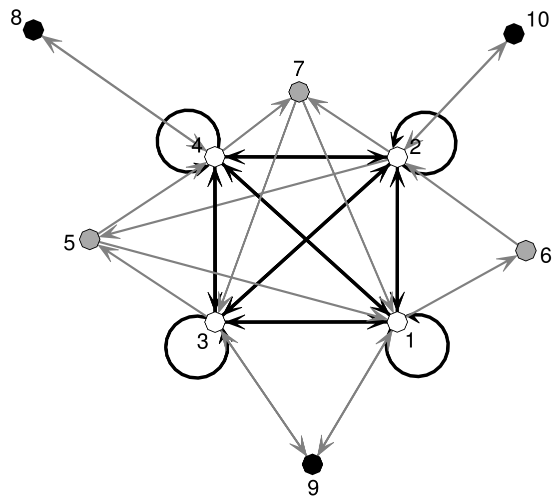

Figure 3: Network 3

	1	2	3	4	8	9	10	5	6	7
1	10	10	10	10		5		5	2	5
2	10	10	10	10			5	2	5	2
3	10	10	10	10		5			2	5
4	10	10	10	10	5			5		2
8				5						
9	5		5							
10		5								
5	5	2		5						
6	2	5	2							
7	5	2	5	2						

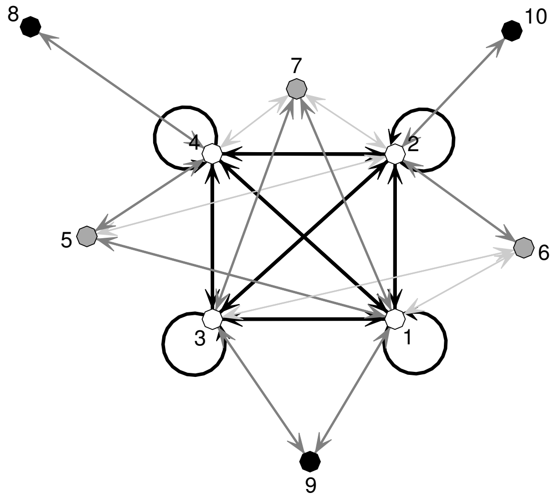

Figure 4: Network 4

	1	2	3	4	8	9	10	5	6	7
1	10	10	10	10		5		5	5	5
2	10	10	10	10			5	5	5	5
3	10	10	10	10		5		5		5
4	10	10	10	10	5			5		5
8				5						
9	5		5							
10		5								
5	5			5						
6		5								
7	5		5							

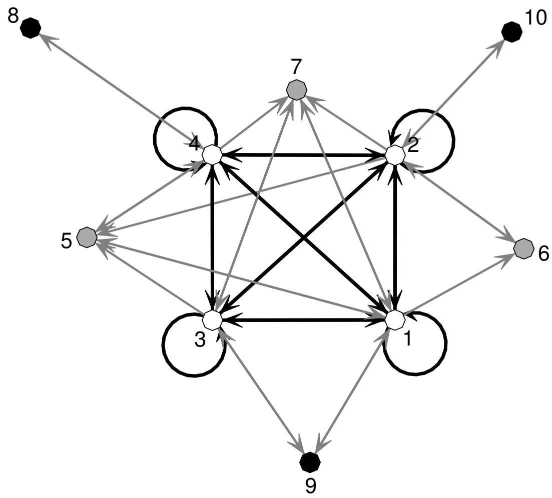

Figure 5: Network of note-borrowing between social-informatics students

	1	2	3	4	5	6	7	8	9	10	11	12	13
1				15				1	8			3	
2			2	3			5	5	10	10	1	3	
3				19				3	1				
4	2		6		1			1	19		1		
5				16		5		7	16		5		3
6			1		4			7	3		7	3	1
7			6	14				14	6				
8				5					6				
9				19				1					
10		16	2	16		1		16			1	2	
11			2	8	2	2		5	14			2	
12	2	2	8	2	2	2	2	2	6	2	11		
13				1	8			8	3				

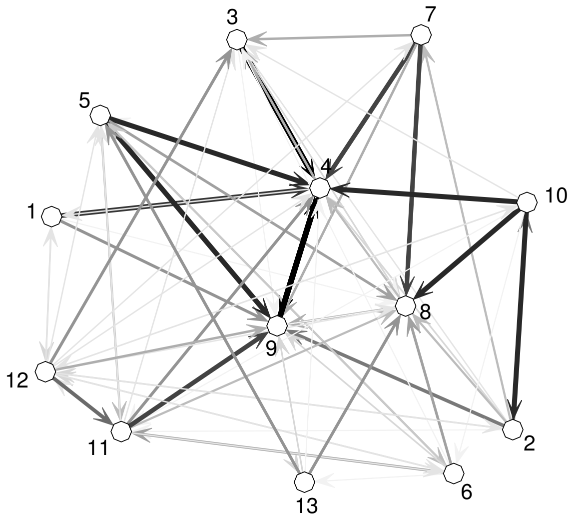

Figure 6: Network of note-borrowing partitioned using binary, valued, and homogeneity blockmodeling

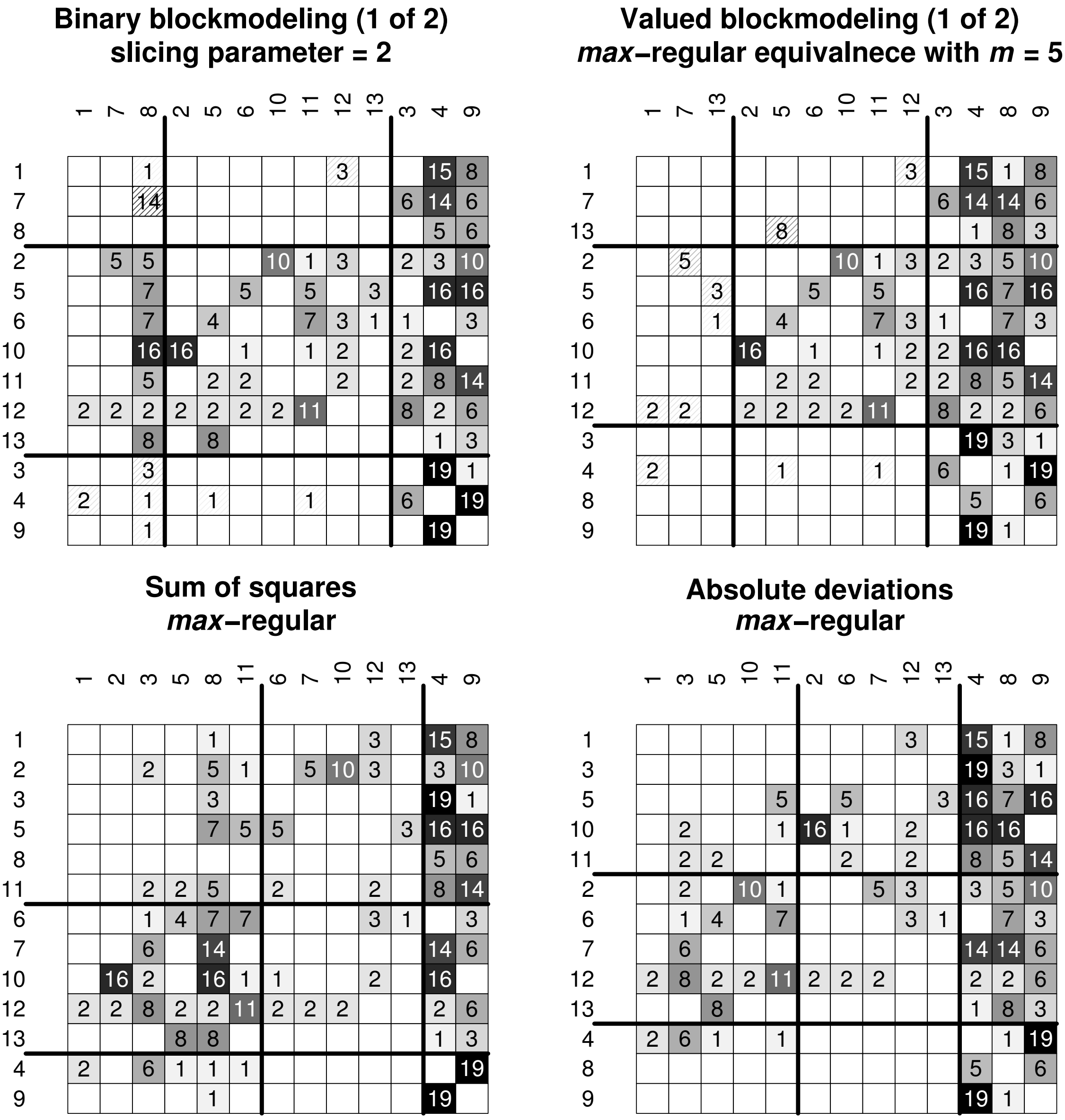

Figure 7: Networks of note-borrowing partitioned using implicit blockmodeling

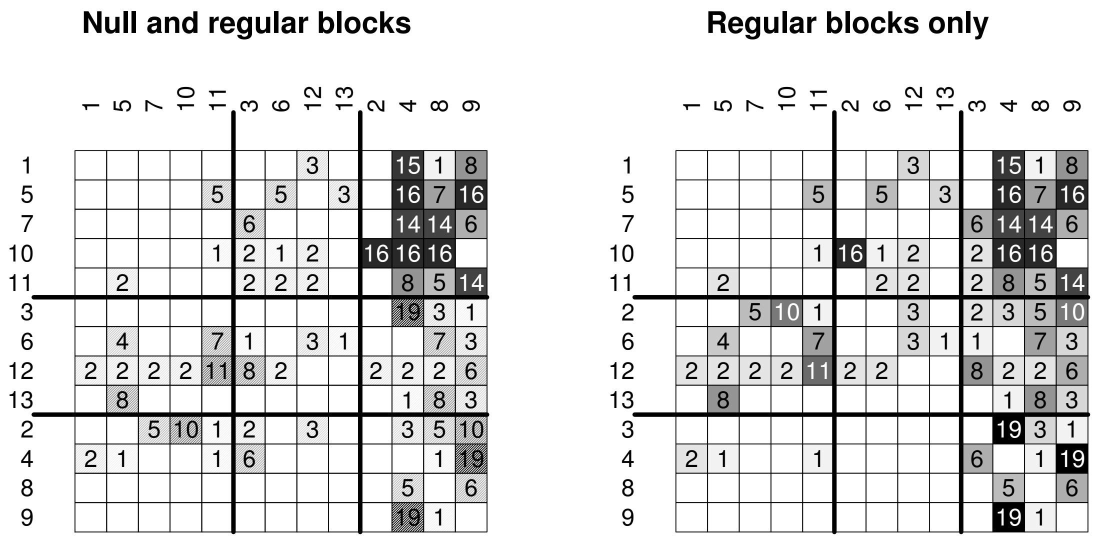

Figure 8: Dendrograms based on REGGE similarities

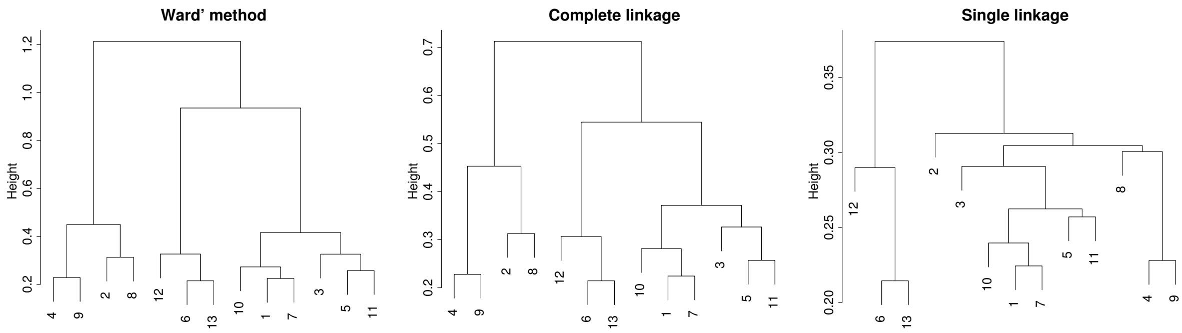

Figure 9: Network of note-borrowing partitioned using different versions of REGE